IN THE CLAIMS:

Please amend the claims as indicated. A complete set of the claims is included below, reflecting added subject matter (underlining) and deleted subject matter (strikethrough), as well as the current status of each claim. This listing of claims will replace all prior versions, and listings, of claims in the application:

1.   (Currently Amended) A ~~computer-implemented~~ method implemented on a computer operating system for scheduling registered services, ~~tasks,~~ the method comprising the steps of:

(a) operating a kernel of **[[a]]** the computer operating system to cycle through a plurality of stored pre-assigned time slices, at least one slice being assigned to a background thread;

(b) scheduling execution of a service manager on the background thread;

(c) ranking the registered services according to priority and resource need;

~~(c)~~ (d) operating the service manager to schedule a plurality of registered services for execution as by rank, each service being scheduled for execution within at least one of the pre-assigned time slices, ~~where~~ whereby the registered services are scheduled for execution independently of any foreground tasks; and

~~(d) ranking the registered services according to priority and resource need; and~~

(e) allocating an execution presence and data presence to a registered service.

2.   (Currently Amended) A method as recited in Claim 1 further comprising ~~the~~ a step wherein ~~of~~ the service manager searches ~~searching~~ for at least one service manager service.

3.   (Currently Amended) A method as recited in Claim 1 wherein the computer operating system comprises ~~method is implemented on~~ a portable electronic device.

4.   (Previously Presented) A method as recited in Claim 1 wherein the data presence is an A5-based global variable context.

5.   (Previously Presented) A method as recited in Claim 1 wherein at least one registered service is a system-related activity.

6.   (Previously Presented) A method as recited in Claim 1 wherein at least one registered service is an interrupt-related activity.

7.   (Previously Presented) A method as recited in Claim 1 wherein at least one registered service is a background-related activity.

8.   (Previously Presented) A method as recited in Claim 1 further comprising the step of periodically repeating the steps (a) through (e).

9.   (Previously Presented) A method as recited in Claim 2 further comprising the step of periodically repeating the step of the service manager searching for at least one service manager service.

10.   (Currently Amended) A ~~computer-implemented~~ method for scheduling services ~~tasks~~ on a computer operating system, comprising the steps of:

a)   performing a background task of registering at least one registered service~~, the task invoked by a kernel of a computer operating system~~ in a dedicated pre-assigned time slice, the task invoked by a kernel of the computer operating system ~~the task~~ for providing an execution presence and a data presence to the registered service;

b)   ~~the task~~ ranking at least one registered service according to requirements of the registered service; and

c)   ~~the task~~ allocating the execution presence and the data presence accordingly to ~~each of~~ the registered services such that ~~each of~~ the registered services ~~is given an opportunity to~~ may be scheduled in [[the]] a dedicated pre-assigned time slice independently of any foreground task.

11.   (Currently Amended) A method as recited in Claim 10 further comprising a [[the]] step wherein [[of]] the background task searches searching for at least one registered service associated therewith.

12.   (Currently Amended) A method as recited in Claim 10 wherein the computer operating system is method is implemented on a portable electronic device.

13.   (Previously Presented) A method as recited in Claim 10 wherein the data presence is an A5-based global variable context.

14.   (Currently Amended) A method as recited in Claim 10 further comprising a [[the]] step of periodically repeating the steps a) through c).

15.   (Currently Amended) A method as recited in Claim 11 further comprising a [[the]] step of periodically repeating the step of background task searching for at least one service associated therewith.

16.   (Currently Amended) A computer system comprising:

a processor coupled to a bus[[;]] and

a memory unit coupled to the bus having stored therein [[an]] a computer operating system for execution by the processor and a background task for execution by the processor [[;]] wherein upon execution the background task, the processor performs a method comprising the steps of:

a)   registering by the background task a plurality of registered services, the background task being invoked by a kernel of [[a]] the computer operating system in a dedicated pre-assigned time slice, the computer operating system operating the background task and a foreground task, the operation of the background task being independent from the operation of the foreground task, and the background task being operated to provide an execution presence and a data presence to a registered service;

b)   ranking the registered services according to priority and resource needs of each registered service; and

c)   allocating the execution presence and the data presence according ~~accordingly~~ to ~~each of~~ the registered services such that ~~each of~~ the registered services ~~is given an opportunity to~~ may be scheduled in the dedicated pre-assigned time slice independently of any foreground task.

17.   (Currently Amended) A computer system as recited in Claim 16 wherein the background task further performs [[the]] a step of searching for the registered services associated with the background task.

18.   (Currently Amended) A computer system as recited in Claim 16 wherein the computer operation system is a portable electronic device.

19.   (Previously Presented) A computer system as recited in Claim 16 wherein the data presence is an A5-based global variable context.

20.   (Previously Presented) A computer system as recited in Claim 16 wherein at least one registered service is a system-related activity.

21.   (Previously Presented) A computer system as recited in Claim 16 wherein at least one registered service is an interrupt-related activity.

22.   (Previously Presented) A computer system as recited in Claim 16 wherein at least one registered service is a background-related activity.

23.   (Currently Amended) A computer-implemented method for scheduling tasks, the method comprising:

a)   cycling through a set of pre-assigned time slices to schedule a set of tasks comprising at least one background task and at least one foreground task, each of the tasks

assigned to one of the time slices wherein scheduling of the background task is independent from
scheduling of the foreground task; and

      b)   scheduling execution of a service manager operating on a background thread
wherein the step b) comprises the steps of:

           b1)   the service manager scheduling a set of tasks that are registered therewith
for execution within the time slice assigned to each task, <u>whereby</u> ~~wherein~~ the set of registered
tasks is dynamically updated; and

           b2)   the service manager allocating a data presence and execution presence to
each of the background tasks registered therewith independently of said foreground tasks.

      24.   (Previously Presented) A method as recited in Claim 23 wherein the method is
implemented on a portable electronic device.

      25.   (Previously Presented) A method as recited in Claim 23 wherein the data
presence is an A5-based global variable context.

      26.   (Currently Amended) A method for scheduling tasks on a computer system <u>in the</u>
<u>course of</u> ~~that is~~ executing a plurality of foreground applications, the method comprising:

      a)   a kernel of an operating system scheduling a plurality of tasks for execution on
the computer system within respective time slices, the plurality of tasks being in a static mode
and one of the tasks being a service manager;

      b)   a plurality of applications dynamically registering with the service manager; and

      c)   the service manager, when itself executing in its time slice, scheduling for
execution the plurality of applications based on a priority and resource need of each application,
wherein applications are scheduled for execution by the service manager in a manner
independent from any of the foreground applications.

      27.   (Previously Presented) A method as recited in Claim 26 wherein the plurality of
applications comprise a system service, an interrupt service and a background service.

28.   (Previously Presented) A method as recited in Claim 26 wherein the computer system is a handheld computer system.

29.   (Previously Presented) A method as recited in Claim 26 wherein the step b) comprises the step of the service manager dynamically registering the plurality of applications based on registration information associated therewith.